CORRECTED VISION

THE ROLE OF CAMERA AND LENS PARAMETERS IN REAL-WORLD MEASUREMENT

JOSEPH HOWSE, NUMMIST MEDIA

CIG-GANS WORKSHOP: 3-D COLLECTION, ANALYSIS AND VISUALIZATION

LAWRENCETOWN, NS: JANUARY 31, 2018

Download me from
http://nummist.com/opencv/Howse_CIG-GANS_20180131.pdf

Here be underscores

MOTIVATING QUESTIONS

1. Are my lens and camera good enough to measure what I want to measure?
2. In software, how can I model the perspective and distortion?
3. Is my computer fast enough to process my camera feed in real time?

- This presentation approaches these questions:
  - Quantitatively
  - Predictively. Recognize feasibility problems at the start of the project, not the end.

MOTIVATING QUESTIONS

1. Are my lens and camera good enough to measure what I want to measure?
2. In software, how can I model the perspective and distortion?
3. Is my computer fast enough to process my camera feed in real time?

Smartphones aren't always the right tool for the job!

- This presentation approaches these questions:
    - Quantitatively
    - Predictively. Recognize feasibility problems at the start of the project, not the end.

OUTLINE

- High-performance imaging
  - Spatial resolution
  - Temporal resolution
- Camera calibration
  - The camera matrix
  - Distortion coefficients
- Computational performance
  - Budgeting operations per pixel per frame
  - Comparing ultra-compact computers

HIGH-PERFORMANCE IMAGING

SPATIAL RESOLUTION

TEMPORAL RESOLUTION

SPATIAL RESOLUTION

- Measurable in line pairs per millimeter (lp/mm)
  - Max density of dark-and-light lines that given combo of lens and sensor can resolve
  - Refers to mm on sensor surface, not subject surface
- combo of lp/mm and magnification determines smallest resolvable detail as measured on subject surface
- Magnification is easy to change by re-focusing lens

SPATIAL RESOLUTION

- Example:
  - A microfilm system using the Zeiss S-Orthoplanar 50mm f/4 lens resolves 360 lp/mm
  - At 1:5 magnification (one-fifth life size), smallest resolvable detail on subject surface is:
    - 1 mm / 360 / (1/5) = 13.9 µm
  - At 1:30 magnification, it is:
    - 1 mm / 360 / (1/30) = 83.3 µm

SPATIAL RESOLUTION

- Limiting factor may be:
  - Diffraction
    - Smaller aperture (larger f-number) is more limiting
    - Longer wavelength of light is more limiting
  - Pixel pitch
    - Distance between photosites; larger is more limiting
  - Lens imperfections
  - Sensor imperfections

SPATIAL RESOLUTION

- $R_{diffraction} = \frac{1}{N\lambda}$, where:
  - $R_{diffraction}$ is the diffraction-limited resolution in lp/mm
  - N is the f-number
  - λ is wavelength of light in mm
    - Human eye's sensitivity peaks at λ = 0.000555 (yellow-green)
- $R_{pitch} = \frac{1}{p}$, where:
  - R_{pitch} is the pixel-pitch-limited resolution in lp/mm
  - p is the pixel pitch in mm

SPATIAL RESOLUTION

- Example:
  - The Nokia Lumia 1020 smartphone has a lens with a maximum aperture of f/2.2 and a sensor with a size of 8.8mm×6.6mm and pixel resolution of 7712×5360
  - $R_{diffraction} = \frac{1}{2.2 \times 0.000555} = 819\ lp/mm$
  - $R_{pitch} = \frac{1}{8.8/7712} = 876\ lp/mm$
  - Conclusion: The high pixel density is an irrational design choice. The resolution is limited theoretically by diffraction and realistically by lens imperfections.

TEMPORAL RESOLUTION

- Things move fast!
  - Waves on the ocean surface
    - Average around 10km/h near shore
  - Cars on the road
  - Conveyor belts in an assembly line
  - Our eyes and eyelids
    - Normal blink lasts 100ms to 400ms

TEMPORAL RESOLUTION

- Faster motion causes problems:
  - The subject appears in fewer frames (before it goes away)
    - Fewer samples to give to detection algorithm
    - Smaller likelihood of detection
  - When the subject does appear, it is blurrier
    - Effectively, less spatial resolution

TEMPORAL RESOLUTION

- Example:
  - Suppose a blink detector's true positive rate is 10% (and its false positive rate is negligible). Each of the subject's blinks lasts 300ms on average.
  - A camera running at 60 FPS captures 18 frames during the average blink. Trying 18 times, the blink detector is $1-(0.9^{18})=85\%$ likely to detect the blink at least once.
  - A camera running at 120 FPS captures 36 frames during the average blink. Trying 36 times, the blink detector is $1-(0.9^{36})=98\%$ likely to detect the blink at least once.

CAMERA CALIBRATION

THE CAMERA MATRIX

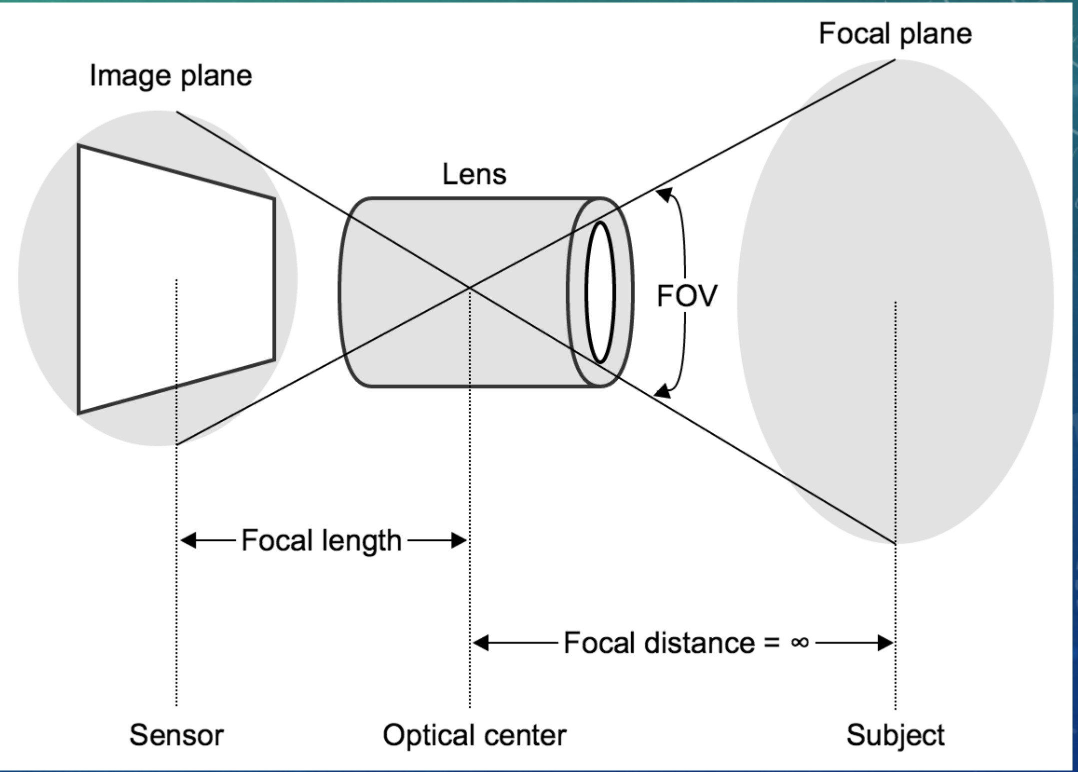
DISTORTION COEFFICIENTS

THE CAMERA MATRIX

The Ideal Camera Matrix		
f	0	$c_x = w/2$
0	f	$c_y = h/2$
0	0	1

- f is focal length
- (c_x, c_y) is center or "principal point" of image within image plane
- (w, h) are width and height of image plane
- (θ, ϕ) are horizontal and vertical field of view (FOV)

$$f = \frac{\sqrt{w^2 + h^2}}{2\sqrt{\left(\tan\frac{\theta}{2}\right)^2 + \left(tan\frac{\phi}{2}\right)^2}}$$



THE CAMERA MATRIX

The Ideal Camera Matrix		
f	0	$c_x = w/2$
0	f	$c_y = h/2$
0	0	1

- f is focal length
- (c_x, c_y) is center or "principal point" of image within image plane
- (w, h) are width and height of image plane
- (θ, ϕ) are horizontal and vertical field of view (FOV)

$$f = \frac{\sqrt{w^2 + h^2}}{2\sqrt{\left(\tan\frac{\theta}{2}\right)^2 + \left(tan\frac{\phi}{2}\right)^2}}$$

- Units must be consistent, e.g.:
  - ✓ f and (cx, cy) are all in mm
  - ✓ Or, f and (cx, cy) are all in pixels
- Spec sheets may give lens's f in mm and image sensor's (w, h) in mm
- Or, APIs may give (θ, ϕ) in degrees or radians and image's (w, h) in pixels
  - Camera API in Android SDK

THE CAMERA MATRIX

The Ideal Camera Matrix		
f	0	$c_x = w/2$
0	f	$c_y = h/2$
0	0	1

- f is focal length
- (c_x, c_y) is center or "principal point" of image within image plane
- (w, h) are width and height of image plane
- (θ, ϕ) are horizontal and vertical field of view (FOV)

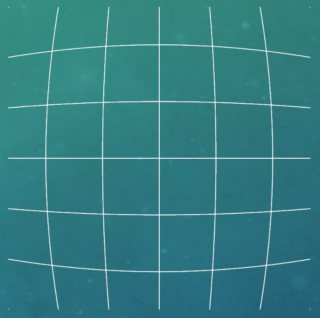
$$f = \frac{\sqrt{w^2 + h^2}}{2\sqrt{\left(\tan\frac{\theta}{2}\right)^2 + \left(tan\frac{\phi}{2}\right)^2}}$$

- f is useful in calculating size or distance
- For ideal lens and camera, $\frac{s_{image}}{f} = \frac{s_{real}}{d}$, where:
  - s_{image} is object's size in image
    - e.g. in pixels, or in mm on sensor surface
  - S_{real} is object's real size
  - d is distance between camera and object

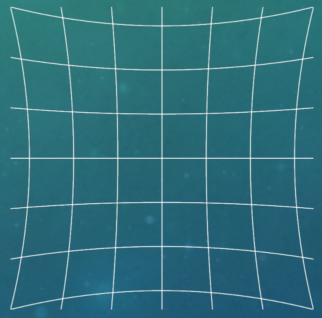
DISTORTION COEFFICIENTS

The Ideal Distortion Coefficients				
$k_1 = 0$	$k_2 = 0$	$p_1 = 0$	$p_2 = 0$	$k_3 = 0$

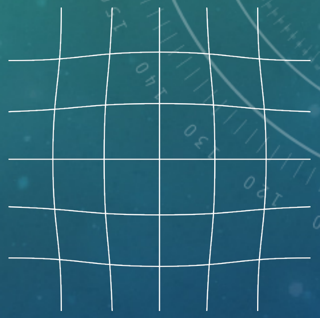
- k_n is the nth radial distortion coefficient
  - $k_1 < 0$ usually implies barrel distortion
  - $k_1 > 0$ usually implies pincushion distortion
  - Changing sign across k_n series may imply moustache distortion
- p_n is the nth tangential distortion coefficient
  - Sign depends on direction of lens's tilt relative to image plane



Barrel Distortion



Pincushion Distortion



Moustache Distortion

DISTORTION COEFFICIENTS

The Ideal Distortion Coefficients				
$k_1 = 0$	$k_2 = 0$	$p_1 = 0$	$p_2 = 0$	$k_3 = 0$

- k_n is the nth radial distortion coefficient
  - $k_1 < 0$ usually implies barrel distortion
  - $k_1 > 0$ usually implies pincushion distortion
  - Changing sign across k_n series may imply moustache distortion
- p_n is the nth tangential distortion coefficient
  - Sign depends on direction of lens's tilt relative to image plane

- Rarely, lens manufacturer may specify distortion coefficients in spec sheets or code samples
- Or, third-party libraries may provide distortion coefficients for various lenses:
  - lensfun: http://lensfun.sourceforge.net
  - Python wrapper, lensfunpy: https://github.com/letmaik/lensfunpy
    - Interoperable with OpenCV and SciPy
- Or, we may have to use calibration process
  - Chessboard calibration in OpenCV

COMPUTATIONAL PERFORMANCE

BUDGETING OPERATIONS PER PIXEL PER FRAME

COMPARING ULTRA-COMPACT COMPUTERS

BUDGETING OPERATIONS PER PIXEL PER FRAME

- Peak performance is often specified in FLOPS: floating point operations per second
  - 1 GFLOPS = 1 billion FLOPS
  - Beware, not all FLOPS are equal!
    - Precision may be half (16-bit), single (32-bit), or double (64-bit)
    - Different architectures have different operations
    - Number of FLOPS in higher-level functions, e.g. in OpenCL, varies depending on drivers

BUDGETING OPERATIONS PER PIXEL PER FRAME

- For a given camera and computer, $b = \frac{p}{v \times w \times h}$, where:
  - b is the budget in floating point operations per pixel per frame
  - p is the computer's peak performance in FLOPS
  - v is the camera's frequency, i.e. the FPS, i.e. the frame rate in Hz
  - (w, h) are the width and height of the image in pixels

BUDGETING OPERATIONS PER PIXEL PER FRAME

- Example:
  - Suppose we capture frames from a Point Grey GS3-U3-23S6C-C camera, with 1920x1200 pixels @ 163 FPS.
  - For an Intel Iris Pro Graphics 580 GPU, capable of 1,152 GFLOPS:
    - $b = \frac{1.152 \times 10^{12}}{163 \times 1920 \times 1200} = 3067$ floating-point operations per pixel per frame
  - For an AMD HD 8210E GPU, capable of 85 GFLOPS:
    - $b = \frac{8.5 \times 10^{10}}{163 \times 1920 \times 1200} = 226$ floating-point operations per pixel per frame

COMPARING ULTRA-COMPACT COMPUTERS: X86

System	Camera Interfaces	CPU	GPU	Peak GPU Performance (Float32)	Peak Power Use*	Price
Intel NUC Kit NUC6i7KYK "Skull Canyon"	USB 3.0 + USB 3.1, Thunderbolt, Ethernet	Quad-core i7-6770HQ	Iris Pro Graphics 580, 72 execution units, OpenCL 2.0, 128 MB eDRAM	1,152 GFLOPS @ 1,000 MHz	85 W	US$595
Gizmo 2	USB 2.0 + USB 3.0, Ethernet	Dual-core Jaguar GX-210HA	HD 8210E, 128 stream processors, OpenCL 1.2	85 GFLOPS @ 300 MHz	9 W	US$199

* Excluding peripherals

COMPARING ULTRA-COMPACT COMPUTERS: X86

System	Camera Interfaces	CPU	GPU	Peak GPU Performance (Float32)	Peak Power Use*	Price
Intel NUC Kit NUC6i7KYK "Skull Canyon"	USB 3.0 + USB 3.1, Thunderbolt, Ethernet	Quad-core i7-6770HQ	Iris Pro Graphics 580, 72 execution units, OpenCL 2.0, 128 MB eDRAM	1,152 GFLOPS @ 1,000 MHz	85 W	US$595
Gizmo 2	USB 2.0 + USB 3.0, Ethernet	Dual-core Jaguar GX-210HA	HD 8210E, 128 stream processors, OpenCL 1.2	85 GFLOPS @ 300 MHz	9 W	US$199

Real-world performance difference in one of my OpenCV applications is 5x but not 13x as GFLOPS specs suggest

* Excluding peripherals

COMPARING ULTRA-COMPACT COMPUTERS: ARM

System	Camera Interfaces	CPU	GPU	Peak GPU Performance (Float32)	Peak Power Use*	Price
NVIDIA Jetson TX2	USB 2.0 + USB 3.0, CSI, Ethernet	Dual-core Denver2 + quad-core Cortex-A57	NVIDIA Pascal, 256 CUDA cores	750 GFLOPS @ 1,465 MHz	15 W	US$599
Odroid-XU4	USB 2.0 + USB 3.0, Ethernet	Quad-core Cortex-A15 + quad-core Cortex-A7	Mali-T624, 6 cores, OpenCL 1.2	142 GFLOPS @ 695 MHz	16 W	US$59

* Excluding peripherals

CONCLUSIONS

- Feasibility assessments should include:
  - Spatial resolution: lp/mm, diffraction, pixel pitch, magnification level
  - Temporal resolution: speed of subject, need for redundancy
    - A detector's "miss rate" decreases exponentially with temporal resolution
  - Camera matrix: availability of data on either focal length or FOV
  - Distortion coefficients: availability of either reference data or run-time calibration results
  - Computational performance: GFLOPS, operations per pixel per frame
- A good lens needs a good camera
- A good camera needs a good processor and good software optimizations

Download me from
http://nummist.com/opencv/Howse_CIG-GANS_20180131.pdf

Here be underscores

QUESTIONS?

JOSEPH HOWSE, NUMMIST MEDIA

HTTP://NUMMIST.COM

JOSEPHHOWSE@NUMMIST.COM